UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/072,693	02/06/2002	Petr Hrebejk	SUN-P6327	2517

24209 7590 06/29/2005

GUNNISON MCKAY & HODGSON, LLP
1900 GARDEN ROAD
SUITE 220
MONTEREY, CA 93940

EXAMINER	
HO, ANDY	
ART UNIT	**PAPER NUMBER**
2194	

DATE MAILED: 06/29/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/072,693	**Applicant(s)** HREBEJK ET AL.
	Examiner Andy Ho	**Art Unit** 2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *08 April 2005*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-43* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-43* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. This action is in response to the amendment filed 4/8/2005.

2. Claims 1-43 have been examined and are pending in the application.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hinson U.S Patent No. 6,829,770.

As to claim 1, Hinson teaches a method for notifying one or more listeners of an event (provides an event class object to distribute information produced by a publisher to one or more subscribers, lines 41-43 column 11) in a object facility repository (142, Fig. 7), the event having an event type (to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13; the type of object to be retrieved from the event objects store 140, lines 60-61 column 16), the method including:

creating an event object for the event (event object 102, Fig. 5; event objects within the storage 142, Fig. 7), the event object corresponding to the event sub-type (class of the event objects having methods, lines 8-41 column 12);

performing the event (firing an event, line 50 column 11);

calling a method for each of the listeners (calling a method on interface 110, lines 50-54 column 11) registered for event notification for the event type (registration for events by the subscribers, lines 40-53 column 13) by passing said event object to listeners (sending the events to the subscribers, line 39 column 11 to line 22 column 12) described in an event source interface corresponding to the event type (COM+ Events system interfaces, lines 35-42 column 16), said method for each of the listeners implemented by each listener (...the subscribers 106-108 are COM Objects that have a subscription to a method or methods of the outgoing-event interface 110, which causes the event class object 102 to propagate events fired on the methods to the subscribers. The subscribers 106-108 individually expose outgoing-event interfaces 116 that are defined identically to the event class object's outgoing-event interface 110. The event class object 102 propagates the event by calling the method of the subscribers' outgoing-event interfaces that correspond to the method called by the publisher in the event class object's interface 110. The implementation of the outgoing-event interface methods in the subscriber 106-108 includes the subscriber code that processes or acts on the event..., lines 49-61 column 12).

Hinson does not explicitly teach the object facility repository is a meta object facility repository. However, Hinson teaches (lines 50-57 column 7) that the invention can be implemented in combination with other program modules that implement particular abstract data types. Therefore one of ordinary skill in the art would conclude that the particular abstract data type could be metadata defining the structure of data objects and the object facility repository of Hinson could be a meta object facility repository.

As to claim 2, Hinson as modified further teaches a listener registers for event notification by passing a registration call to a class implemented by the MOF designed to track listener registrations in said event source interface corresponding to the event type (...the COM+ Events system 140 includes services in the form of application programming interface functions that can be called to register, remove or modify subscriptions, as well as to retrieve or enumerate subscriptions to a particular outgoing-event interface method; the subscriber objects 106-108 implement the code that calls the COM+ Events API function for registering a subscription so as to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13).

As to claim 3, Hinson as modified further teaches class implemented by the MOF corresponds only to the event type (to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13).

As to claim 4, Hinson as modified further teaches class implemented by the MOF maintains a record of which listener is registered for notification of which events (stores persistent subscriptions, lines 34-39 column 13).

As to claim 5, Hinson as modified further teaches the event type is a class, instance, or association (class of the event objects having methods, lines 8-41 column 12; to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13).

As to claim 6, Hinson as modified further teaches calling a method for each of the listeners (calling a method on interface 110, lines 50-54 column 11) registered for event notification for the event sub-type (...the subscriptions specify which subscribers to call whenever a call is made to the subscribed event class method. The subscribers also expose an interface matching the event class interface that contains the subscribed method..., lines 25-30 column 4; class of the event objects having methods, lines 8-41 column 12; registration for events by the subscribers, lines 40-53 column 13) by passing the event object to event sub-type listeners (sending the events to the subscribers, line 39 column 11 to line 22 column 12) as indicated by a bitmask in a combination event source interface (...subscription 120 is an object that contains an event class identifier 122 and an event method identifier 124 to specify the events, and a subscriber reference 126 to specify the subscriber 106..., line 67 column 12 to line 3 column 13), the method for each of the listeners registered for event notification implemented by each listener (...the subscribers 106-108 are COM Objects that have a subscription to a method or methods of the outgoing-event interface 110, which causes the event class object 102 to propagate events fired on the methods to the subscribers. The subscribers 106-108 individually expose outgoing-event interfaces 116 that are defined identically to the event class object's outgoing-event interface 110. The event class object 102 propagates the event by calling the method of the subscribers' outgoing-event interfaces that correspond to the method called by the publisher in the event class object's interface 110. The implementation of the outgoing-event interface methods in the subscriber 106-108 includes the subscriber code that processes or acts on the event..., lines 49-61 column 12), wherein a listener registers for event notification for an event type by setting a bit corresponding to the event sub-type in a combination event source interface (contains an event class identifier 122 and an event method identifier 124 to specify the events, line 67 column 12 to line 3 column 13).

As to claim 7, Hinson as modified further teaches calling a method for each of the listeners registered for vetoable event notification for the event type by passing said event object to listeners described in an event source interface corresponding to the event type, said method for each of the listeners registered for vetoable event notification implemented by each listener and ending said method for notifying before said performing the event if a veto is received by any of said listeners registered for vetoable event notification (the use of subscriber filter, line 62 column 20 to line 53 column 21; ...on delivery of an event on a subscription with a subscriber filter, the event class object 102 first calls the subscriber filter's "ISubscriberControl::PreEventCall( )" method before firing the event as a call to the subscriber filter's outgoing-event interface 110..., lines 9-13 column 21).

As to claim 8, it is a method claim of claims 2 and 7. Therefore, it is rejected for the same reasons as claims 2 and 7 above.

As to claims 9-11, they are method claims of claims 3-4 and 6, respectively. Therefore, they are rejected for the same reasons as claims 3-4 and 6 above.

As to claim 12, it is a method claim of claims 1 and 6. Therefore, it is rejected for the same reasons as claims 1 and 6 above.

As to claim 13, Hinson teaches a method for registering for event notification of an event (registration for events by the subscribers, lines 40-53 column 13) in an object facility repository (142, Fig. 7), the event having an event type (to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13; the type of object to be retrieved from the event objects store 140, lines 60-61 column 16), the method including:

implementing a method for event notification corresponding to the event type (method on interface 110, lines 50-54 column 11);

passing a registration call to a class implemented by the MOF designed to track listener registrations in an event source interface corresponding to the event type (...the COM+ Events system 140 includes services in the form of application programming interface functions that can be called to register, remove or modify subscriptions, as well as to retrieve or enumerate subscriptions to a particular outgoing-event interface method; the subscriber objects 106-108 implement the code that calls the COM+ Events API function for registering a subscription so as to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13).

Hinson does not explicitly teach the object facility repository is a meta object facility repository. However, Hinson teaches (lines 50-57 column 7) that the invention can be implemented in combination with other program modules that implement particular abstract data types. Therefore one of ordinary skill in the art would conclude that the particular abstract data type could be metadata defining the structure of data objects and the object facility repository of Hinson could be a meta object facility repository.

As to claim 14, Hinson teaches a method for registering for event notification of an event (registration for events by the subscribers, lines 40-53 column 13) in an object facility repository (142, Fig. 7), the event having an event type (to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13; the type of object to be retrieved from the event objects store 140, lines 60-61 column 16) and event sub-type (class of the event objects having methods, lines 8-41 column 12), the method including:

implementing a method for event notification corresponding to a combination event type (method on interface 110, lines 50-54 column 11);

passing a bitmask (...subscription 120 is an object that contains an event class identifier 122 and an event method identifier 124 to specify the events, and a subscriber reference 126 to specify the subscriber 106..., line 67 column 12 to line 3 column 13) to a class implemented by the MOF designed to track listener registrations (...the COM+ Events system 140 includes services in the form of application programming interface functions that can be called to register, remove or modify subscriptions, as well as to retrieve or enumerate subscriptions to a particular outgoing-event interface method; the subscriber objects 106-108 implement the code that calls the COM+ Events API function for registering a subscription so as to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13) in a combination event type source interface (sending the events to the subscribers, line 39 column 11 to line 22 column 12), the bitmask indicating on which event sub-types to receive event notification (...the subscriptions specify which subscribers to call whenever a call is made to the subscribed event class method. The subscribers also expose an interface matching the event class interface that contains the subscribed method..., lines 25-30 column 4; class of the event objects having methods, lines 8-41 column 12; registration for events by the subscribers, lines 40-53 column 13).

Hinson does not explicitly teach the object facility repository is a meta object facility repository. However, Hinson teaches (lines 50-57 column 7) that the invention can be implemented in combination with other program modules that implement particular abstract data types. Therefore one of ordinary skill in the art would conclude that the particular abstract data type could be metadata defining the structure of data objects and the object facility repository of Hinson could be a meta object facility repository.

As to claim 15, Hinson teaches an apparatus for notifying one or more listeners of an event (provides an event class object to distribute information produced by a publisher to one or more subscribers, lines 41-43 column 11) in an object facility repository (142, Fig. 7), the event having an event type (to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13; the type of object to be retrieved from the event objects store 140, lines 60-61 column 16) and sub-type (class of the event objects having methods, lines 8-41 column 12), the apparatus including:

an event source interface corresponding to the event type (COM+ Events system interfaces, lines 35-42 column 16); and a memory storing an event object (event objects within storage 142, Fig. 7), said event object corresponding to the event sub-type (class of the event objects having methods, lines 8-41 column 12), said memory coupled to said event source interface (connection of 140 and 142, Fig. 7).

Hinson does not explicitly teach the object facility repository is a meta object facility repository. However, Hinson teaches (lines 50-57 column 7) that the invention can be implemented in combination with other program modules that implement particular abstract data types. Therefore one of ordinary skill in the art would conclude that the particular abstract data type could be metadata defining the structure of data objects and the object facility repository of Hinson could be a meta object facility repository.

As to claims 16-20, they are apparatus claims of claims 1, 4, 8, 3 and 7, respectively. Therefore, they are rejected for the same reasons as claims 1, 4, 8, 3 and 7 above.

As to claim 21, Hinson teaches an apparatus for notifying one or more listeners of an event (provides an event class object to distribute information produced by a publisher to one or more subscribers, lines 41-43 column 11) in an object facility repository (142, Fig. 7), the event having an event type (to subscribe to a particular outgoing-event interface method..., lines 40-53 column 13; the type of object to be retrieved from the event objects store 140, lines 60-61 column 16) and an event sub-type (class of the event objects having methods, lines 8-41 column 12), the apparatus including:

a combination event type source interface (COM+ Events system interfaces, lines 35-42 column 16); and

a memory storing an combination event object (event objects within storage 142, Fig. 7; ...subscription 120 is an object that contains an event class identifier 122 and an event method identifier 124 to specify the events, and a subscriber reference 126 to specify the subscriber 106..., line 67 column 12 to line 3 column 13), said memory coupled to said combination event type source interface (connection of 140 and 142, Fig. 7).

Hinson does not explicitly teach the object facility repository is a meta object facility repository. However, Hinson teaches (lines 50-57 column 7) that the invention can be implemented in combination with other program modules that implement particular abstract data types. Therefore one of ordinary skill in the art would conclude that the particular abstract data type could be metadata defining the structure of data objects and the object facility repository of Hinson could be a meta object facility repository.

As to claims 22-23, they are apparatus claims of claims 6-7, respectively. Therefore, they are rejected for the same reasons as claims 6-7 above.

As to claims 24-25, they are apparatus claims of claims 13-14, respectively. Therefore, they are rejected for the same reasons as claims 13-14 above.

As to claims 26-39, they are apparatus claims of claims 1-14, respectively. Therefore, they are rejected for the same reasons as claims 1-14 above.

As to claims 40-43, they are computer program product claims of claims 1 and 12-14, respectively. Therefore, they are rejected for the same reasons as claims 1 and 12-14 above.

Response to Arguments

4. Applicant's arguments filed 4/8/2005 have been fully considered but they are not persuasive.

Applicant argued that the examiner have not cited any portion from Hinson reference that shows objects relationship that could interpreted as a layered architecture (Remarks, last incomplete paragraph page 15 to first incomplete paragraph page 16). In response, the applicant argued limitations that are not claimed.

Applicant argued that Hinson reference does not teach a meta object facility repository. The applicant further argued the characteristic of the meta object facility repository as disclosed in the application's specification (Remarks, first complete paragraph page 16 to last complete paragraph page 17). In response, the claim language simply claims an event notification system wherein the event listeners register with the system for a particular event type. The claims further disclose that the notification system is in a meta object facility repository. However, how the events and objects make up a layered metadata architecture is not brought out in the claim language. The reference meets the limitation as claimed.

Applicant argued that Hinson reference does not teach a bitmask (Remarks, first complete paragraph page 18). In response, as clearly disclosed in the claim rejection above, Hinson teaches passing the event object to event sub-type listeners (sending the events to the subscribers, line 39 column 11 to line 22 column 12) as indicated by a bitmask in a combination event source interface (...subscription 120 is an object that contains an event class identifier 122 and an event method identifier 124 to specify the events, and a subscriber reference 126 to specify the subscriber 106..., line 67 column 12 to line 3 column 13), the method for each of the listeners registered for event notification implemented by each listener (...the subscribers 106-108 are COM Objects that have a subscription to a method or methods of the outgoing-event interface 110, which causes the event class object 102 to propagate events fired on the methods to the subscribers. The subscribers 106-108 individually expose outgoing-event interfaces 116 that are defined identically to the event class object's outgoing-event interface 110. The event class object 102 propagates the event by calling the method of the subscribers' outgoing-event interfaces that correspond to the method called by the publisher in the event class object's interface 110. The implementation of the outgoing-event interface methods in the subscriber 106-108 includes the subscriber code that processes or acts on the event..., lines 49-61 column 12), wherein a listener registers for event notification for an event type by setting a bit corresponding to the event sub-type in a combination event source interface (contains an event class identifier 122 and an event method identifier 124 to specify the events, line 67 column 12 to line 3 column 13). The reference meets the limitation as claimed.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andy Ho whose telephone number is (571) 272-3762. A voice mail service is also available for this number. The examiner can normally be reached on Monday – Friday, 8:30 am – 5:00 pm.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 571-272-2100.

Any response to this action should be mailed to:

Commissioner for Patents
P.O Box 1450
Alexandria, VA 22313-1450

Or fax to:

- AFTER-FINAL faxes must be signed and sent to (703) 872 - 9306.
- OFFICAL faxes must be signed and sent to (703) 872 - 9306.
- NON OFFICAL faxes should not be signed, please send to (571) 273 – 3762

A.H
June 23, 2005

ST. JOHN COURTENAY III
PRIMARY EXAMINER